



**IOT-ANTURIDATAN KERÄÄMINEN, KÄSITTELY
JA VISUAALINEN ESITTÄMINEN HYÖDYNTÄEN
MICROSOFT AZUREA JA POWER BI
RAPORTOINTIPALVELUA**

Teemu Saukkosaari

Ohjaaja: Juha-Pekka Mäkelä

SÄHKÖTEKNIIKAN TUTKINTO-OHJELMA

2019

Saukkosaari T. (2019) IoT-anturidatan kerääminen, käsittely ja visuaalinen esittäminen hyödyntäen Microsoft Azurea ja Power BI raportointipalvelua. Oulun yliopisto, Elektroniikan ja tietoliikennetekniikan tutkinto-ohjelma. Kandidaatintyö, 33 s

TIIVISTELMÄ

Tässä tutkielmassa on esitelty ratkaisu simuloidun anturidatan lähettämiseksi Microsoft Azure -pilvipalveluun käyttäen Raspberry Pi -minitietokonetta. Anturidata lähetetään Node-RED ohjelmointityökalun avulla Azure IoT-Hubille, josta se ohjataan talteen alustan taulukkovarastoon. Lopuksi data haetaan ja visualisoidaan Microsoft Power BI -ohjelmistolla.

Esitetty ratkaisu saatiin toimimaan halutulla tavalla ja sitä voidaan tulevaisuudessa laajentaa hyvin suurenkin luokan anturidatan käsittelyyn. Tätä järjestelmää hyödyntäen ja jalostaen Oulun yliopiston Fab Lab voisi kehittää ratkaisun kaikkien laitteiden käyttöhistorian tarkasteluun, sekä laajentaa sen myös kattamaan reaaliaikaisen datan tarkastelun.

Avainsanat: esineiden internet, laitteisto, pilvipalvelu, Raspberry Pi, Node-RED, Microsoft Azure, Microsoft Power BI

Saukkosaari T. (2019) Gathering, Processing and Visually Presenting IoT Sensor Data Using Microsoft Azure and Power BI Reporting Service. University of Oulu, Degree Program in Electrical Engineering, Bachelor's thesis, 33 p.

ABSTRACT

This thesis introduces a solution for sending generated sensor data to Microsoft Azure cloud platform using Raspberry Pi minicomputer. The data is sent by Node-RED programming tool to Azure IoT Hub, from where it is then routed and stored to platform's table storage. Lastly, the historical data is retrieved and visually presented using Microsoft Power BI.

The proposed solution was successfully implemented, and in future it can be extended for processing much larger sensor data. Fab Lab of Oulu University could further develop the solution to include usage history data of all devices as well as to perform real time data analysis.

Key words: internet of things, hardware, cloud service, Raspberry Pi, Node-RED, Microsoft Azure, Microsoft Power BI

SISÄLLYS

TIIVISTELMÄ.....	2
ABSTRACT	3
SISÄLLYS	4
ALKULAUSE	5
1. Johdanto.....	6
2. Esineiden internet.....	7
2.1. Sovelluskohteet.....	8
2.2. Alustat	8
2.2.1. Laitteistoalustat	9
2.2.2. Pilvipalvelualustat	10
3. Anturidatan keräys, käsittely ja visualisointi.....	12
3.1. Raspberry Pi ja Node-RED	12
3.2. Microsoft Azure	15
3.3. Microsoft Power BI.....	17
4. Yhteenveto	21
5. Lähteet	22
6. Liitteet	24

ALKULAUSE

Tämä kandidaatin tutkielma on tehty Oulun yliopiston tietoliikennetekniikan osastolle. Tutkielman tekeminen aloitettiin syksyllä 2018. Haluan kiittää ohjaajaani Juha-Pekka Mäkelää kaikesta avusta työhön liittyen.

Oulussa, lokakuussa 2019

Teemu Saukkosaari

1. JOHDANTO

IoT (Internet of Things) eli suomeksi esineiden internet tarkoittaa erilaisten esineiden ja laitteiden yhdistämistä internetiin sekä toisiin esineisiin ja laitteisiin. Esineet keräävät tietoa ja ovat yhteydessä ulkomaailmaan niihin kytkettyjen antureiden ja toimilaitteiden avulla sekä kommunikoivat toistensa kanssa erilaisia tiedonsiirtotekniikoita hyödyntäen. Tämä mahdollistaa arkipäiväistenkin esineiden muuttamisen älykkäiksi helpottaen ihmisten elämää. Tämä mahdollistaa lisäksi järjestelmät, jotka voivat muun muassa vähentää kustannuksia teollisuudessa, edistää terveydenhoitopalveluja sekä nopeuttaa pelastustehtäviä. Esimerkiksi Iso-Britanniassa pelastuslaitos hyödyntää IoT-järjestelmää, joka toimii erilaisilla antureilla. Muun muassa ajoneuvojen varustukseen liitettyjen RFID-antureiden avulla he varmistavat, että tarvittava pelastuskalusto on mukana. Vesianturien avulla he näkevät mukana olevan veden määrän, sekä paljonko sitä kului tehtävän aikana. Kaiken kerätyn datan avulla tehtäväraportti on helppo täyttää, sillä he näkevät järjestelmästä ketä tehtävällä on ollut mukana ja kuinka kauan se kesti. Kaiken tämän ansiosta pelastuslaitoksen henkilöstö voi keskittyä paremmin heidän tärkeimpään tehtäväänsä eli henkien pelastamiseen. [1]

Tässä kandidaatintyössä kehitettiin IoT-ratkaisua Oulun yliopiston Fab Labiin. Tavoitteena oli saada laitteet kuten 3D-tulostin tai laser leikkuri tallentamaan anturidataa pilvipalveluun. Tarkoituksena oli, että erilaisten anturien avulla voidaan tarkastella laitteiden käyttöhistoriaa jälkeenpäin. Ratkaisuun sisältyi myös datan visuaalinen esitys, jolla sen tarkastelusta tehtiin selkeämpää. Työ mahdollistaa tulevaisuudessa laitteiden monitoroinnin ja pilvipohjaisten IoT ratkaisuiden kehittämisen Fab Labissa.

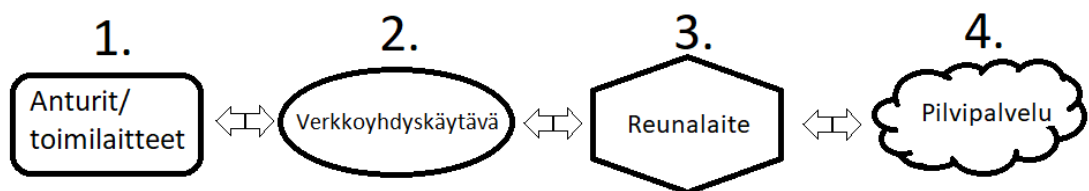
2. ESINEIDEN INTERNET

Esineiden internet on laaja käsite, joka viittaa järjestelmään, jossa yhdistetään erilaisia esineitä kommunikoimaan keskenään verkon yli. Järjestelmään kuuluvat esineet keräävät niihin kytkettyjen anturien avulla dataa, jonka jälkeen tämä data voidaan käsitellä ja analysoida käyttökohteen kannalta oleellisella tavalla. Tällä saavutetaan kehittyneitä ja älykkäitä järjestelmiä palvelemaan ihmisiä. [2]

IoT:lle ei ole selvää yksittäistä määritelmää, mutta IEEE (Institute of Electrical and Electronics Engineers) on kuitenkin rajannut käsitettä ainakin seuraavilla kriteereillä:

- Järjestelmässä täytyy olla yhteys esineiden välillä.
- Esineet ovat yhteydessä internetiin.
- Jokainen järjestelmässä oleva esine on yksilöllisesti tunnistettavissa.
- Järjestelmään pääsee sisälle tapauskohtaisesti mistä ja milloin vain.
- Esineet reagoivat ympäristöönsä niihin kytketyillä erilaisilla antureilla tai laitteilla.
- Älykkäät esineet toimivat työkaluina helpottaamaan ihmisten työskentelyä.
- Järjestelmällä on kommunikointikyky, joka perustuu standardiin ja kommunikaatioprotokoliin.
- Järjestelmä osaa automaattisesti mukautua, esimerkiksi muuttaa energiankulutusta tarpeen mukaan.
- Järjestelmä on ohjelmoitavissa eli sen käyttäytymiseen voidaan vaikuttaa ilman fyysisiä muutoksia. [3]

Kuvassa 1 havainnollistetaan neljän tason IoT-mallia, joka toimii hyvin yksinkertaisena pohjana erilaisille järjestelmille. Ensimmäisessä tasossa esineisiin liitetyt anturit ja toimilaitteet keräävät dataa tai muokkaavat olosuhteita, jotka tuottavat dataa. Tämä data on analogista ja toiselle tasolle siirtyessä se muutetaan digitaalseksi. Toinen taso koostuu verkkoyhdyskäytävästä, joka sijaitsee usein ensimmäisen tason antureiden ja toimilaitteiden läheisyydessä. Verkkoyhdyskäytävä voidaan toteuttaa joko laitteistolla tai ohjelmistolla. Esineeseen voidaan esimerkiksi antureiden lisäksi kytkeä laite, joka toimii verkkoyhdyskäytävänä keräten antureilta saadun datan lähettäen sen kolmannelle tasolle eli reunalaitteelle. Reunalaite sijaitsee tyypillisesti myös samassa paikassa aikaisempien tasojen kanssa, esimerkiksi saman tehtaan sisällä. Reunalaitteen tärkeimpiä tehtäviä on analysoida, prosessoida ja suodattaa dataa, jolloin vain oleellinen osa siitä jatkaa neljännelle tasolle pilvipalveluun. Pilvipalvelujärjestelmän suuremman suorituskyvyn ansiosta dataa voidaan analysoida ja hallinnoida entistä perusteellisemmin. Lisäksi palvelua voidaan käyttää datan säilömiseen. [4]



Kuva 1. Neljän tason yksinkertainen IoT-malli.

2.1. Sovelluskohteet

Sovelluskohteiden näkökulmasta IoT-järjestelmät voidaan jakaa karkeasti kolmeen eri luokkaan, joista ensimmäinen on valvonta ja hallinta. Tähän luokkaan kuuluva järjestelmä valvoo reaaliajassa laitteiden suorituskykyä, energiankulutusta ja ympäristöä. Tämän ansiosta voidaan huomata mahdolliset kehityskohteet, optimoida toimintaa ja ennakoida tuloksia, jolloin kulut pienentyvät ja tuottavuus nousee. Esimerkiksi älykoti on yksi monista järjestelmistä, joka voidaan sijoittaa tähän luokkaan. Siihen liitettyä turvajärjestelmää, ilmastointia ja valaistusta voidaan hallita ja valvoa etänä tietokoneella, puhelimella tai tabletilla. Tämän avulla voitaisiin esimerkiksi sammuttaa päälle jääneet valot kodin ulkopuolelta. [5]

Massadata ja analysointi on IoT-järjestelmien toinen luokka. Anturit ja toimilaitteet keräävät suuret määrät dataa, jota analysoidaan erilaisilla työkaluilla. Analysoitua dataa voitaisiin hyödyntää esimerkiksi liiketoiminnassa selvittämään asiakkaiden käyttäytymistä, jonka avulla parannettaisiin heidän tyytyväisyyttään. Massadataa ja sen analysointia voidaan käyttää lisäksi myös terveydenhoitoalalla. Potilaista voitaisiin esimerkiksi terveydenhoidossa kerätä ja analysoida dataa, jonka ansiosta hoitohenkilökunnan resurssit pystyttäisiin jakamaan paremmin tarpeen mukaan. [5]

Kolmas luokka on tiedon jakaminen ja vuorovaikutus. Vuorovaikutusta voi tapahtua sekä esineiden, ihmisten ja esineiden että ihmisten välillä. Tätä voidaan hyödyntää toimitusketjuissa parantamaan tilannetietoisuutta ja välttämään tiedon viivästymistä. Järjestelmää voitaisiin soveltaa esimerkiksi kauppojen kylmäaltaisiin, jolloin laitteen mennessä epäkuntoon voisi myymäläpäällikkö heti tarkistaa järjestelmästä työntekijöiden tilanteen. Tämän jälkeen päällikkö voisi lähettää tehtävän vapaalle työntekijälle järjestelmään liitetyn laitteen avulla. [5]

IoT-järjestelmät voivat olla hyvinkin erilaisia, jonka ansiosta myös sovelluskohteita on paljon. Muita mahdollisia sovelluskohteita voivat olla esimerkiksi liikenteessä kuljetusten seuranta ja reittiohjaus sekä maataloudessa kastelujärjestelmien automatisointi ja karjan seuranta. Teollisuudessa prosessia voidaan valvoa ja ohjata sekä laitteiden kuntoa tarkastella, jolloin huollot tiedetään ajoittaa oikeaan aikaan. Usein järjestelmät ovat jonkinlaisia yhdistelmiä edellä mainituista kolmesta luokasta.

2.2. Alustat

IoT-järjestelmiin sisältyy usein sekä laitteisto- että pilvipalvelualusta. Alustojen valintaan vaikuttavat järjestelmän koko ja monimutkaisuus, jolloin täytyy pohtia tarvittavan suorituskyvyn ohella myös kustannuksia. Lisäksi laitteistoa valitessa täytyy ottaa huomioon energiankulutus ja verkkoyhteyden laatu. Järjestelmä voi esimerkiksi vaatia laitteiston toimimaan akulla ja langattomalla verkkoyhteydellä. Jos yhteys tukiasemaan on heikko, voi laitteen virrankulutus nousta kasvaneesta lähetystehotarpeesta ja järjestelmän viive kohota uudelleen lähetyksiin kuluva ajasta johtuen [6]. Nämä seikat täytyy huomioida järjestelmää suunniteltaessa.

2.2.1. Laitteistoalustat

IoT:iin soveltuvia eri laitteistoja on paljon, jonka vuoksi niistä esitellään lyhyesti vain muutama. Yksi tunnetuimmista on Arduino, joka on mikrokontrollerialusta ja se soveltuu esimerkiksi anturidatan keräämiseen esineiltä. Kuvan 1 mallissa se voi toimia ensimmäisen tason lisäksi myös toisessa tasossa toimien verkkoyhdyskäytävänä. Sen parhaimpiin puoliin kuuluu sen valtava yhteisö, jossa jaetaan esimerkiksi ohjeita erilaisiin projekteihin aloittelijaystävällisellä askel-askeleelta -periaattella.

Arduino perustuu avoimeen laitteistoon ja lähdekoodiin. Sen kehitysympäristö toimii niin Windowsilla, Macintosh OSX:llä kuin Linuxillakin ja ohjelmointi tapahtuu C++:aan pohjautuvalla kielellä. Arduinon eri malleja on monia, jotka soveltuvat erilaisiin tehtäviin. Ne eroavat toisistaan muun muassa fyysisessä koossa, suorituskvyssä sekä kytkentäpinnien ja porttien lukumäärissä. Lisäksi alustaan on mahdollista liittää erilaisia moduuleita, jotka lisäävät sen ominaisuuksia. Esimerkiksi WiFi-moduulilla saadaan langattoman verkkoyhteyden tuki malliin, jossa sitä ei alunperin ole integroituna. Hinnat vaihtelevat laajasti eri mallien välillä, mutta esimerkiksi Arduinon IoT Bundleen sisältyvä WiFi-tuella varustettu MKR1000-alusta maksaa Arduinon omassa kaupassa 31,99 €. [7]

SensorTag on Texas Instrumentsin referenssialusta cc2650 Bluetooth mikrokontrollerille ja erinäisille antureille. Sen tuotantoon voidaan valita soveltuvat osat, jonka ansiosta sen integroiminen IoT-järjestelmään toimii hyvin. Texas Instruments myy myös valmiita pakkauksia; esimerkiksi SimpleLink multi-standard CC2650STK SensorTag kit tukee valmiiksi kymmentä erilaista integroitua anturia, joihin sisältyy esimerkiksi valo-, kosteus-, paine- ja lämpötila-anturit. Tämän lisäksi siihen voidaan vielä lisätä muita antureita ja toimilaitteita. Mikrokontrollerina siinä toimii langaton ARM Cortex-M3 CC2650, joka on tarkoitettu Bluetooth-, ZigBee ja 6LoWPAN -tekniikoille. Alustan energiankulutusta mainostetaan jopa niin alhaiseksi, että virtalähteenä käytetyllä yhdellä nappiparistolla alusta toimii useita vuosia. Kuvan 1 kokonaisuuteen SensorTag asettuu Arduinon tapaan ensimmäiselle ja toiselle tasolle. SimpleLink multi-standard CC2650STK SensorTag kit maksaa Texas Instrumentsin sivuilta ostettuna 29.00 \$, joka on tällä hetkellä euroiksi muutettuna noin 26,15 €. [8]

Raspberry Pi on yhden piirilevyn minitietokone. Sen on kehittänyt The Raspberry Pi Foundation tarkoituksenaan antaa kaikille mahdollisuus edulliseen tietokoneeseen. Esimerkiksi Raspberry Pi 3 B+ on varustettu 1.4 GHz:n neliytimisellä Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit -suorittimella. Siinä on 1 GB:n keskusmuisti ja integroidut WLAN- (Wireless Local Area Network) ja Bluetooth-piirit. WLAN tukee 2.4 GHz:n ja 5 GHz:n kanavia 802.11.b/g/n/ac -standardilla. Verkkoyhteys voidaan myös muodostaa tietokoneen Ethernet-portin kautta. Lisäksi siinä on hiirelle, näppäimistölle ja muille oheislaitteille neljä USB 2.0 -porttia ja näytölle HDMI-portti sekä esimerkiksi antureille 40-pinninen GPIO (General Purpose Input/Output)-liitinrima. Tämä tietokone tarvitsee 5 voltin käyttöjännitteen ja sen virrankulutus ilman ulkoisia laitteita ja antureita on noin 700-1000 mA. Virtalähteeksi siihen suositellaan 5 V/2.5 A DC -laturia, joka voidaan kytkeä laitteen micro-USB -porttiin. Laitteen massamuistia varten tietokoneesta löytyy micro SD -muistikorttipaikka. IoT-käytössä laite soveltuu hyvin esimerkiksi datan esikäsittelyyn ja analysointiin suorituskvyksensä ansiosta, jolloin sitä voidaan hyödyntää kuvan 1 mallissa ensimmäisen ja toisen tason lisäksi myös kolmannella tasolla. Raspberry Pi

Foundationilla ei ole omaa verkkokauppaa, mutta Suomessa 3 B+ -malli maksaa noin 35 - 45 €. [9]

2.2.2. Pilvipalvelualustat

IoT-käyttöön tarkoitetuilla pilvipalvelualustoilla on sama päämäärä; toimia väylänä yhdistäen eri laitteet ja sovellukset. Eri alustojen ominaisuuksien ja helppokäyttöisyyden vertailu on kuitenkin haastavaa ilman käytännön kokeilua. Myöskään hintojen vertailu ei ole yksinkertaista. Kustannukset alustoilla perustuvat datan määrään tai yksittäisien viestien lukumäärään. Lisäksi erilaiset alustalla käytetyt resurssit voivat lisätä hintaa. Mahdollisuutena voi olla myös esimerkiksi kiinteä kuukausimaksu käytetyistä palveluista tai hinta kulutuksen mukaan. [10]

Pilvipalvelualustoja on satoja erilaisia, joista seuraavaksi esitellään lyhyesti vain kolme suosituinta. Amazon Web Services (AWS) on yksi ensimmäisistä julkisista pilvipalvelualustoista ja se on johtanut alustojen välistä kilpailua jo yli 10 vuotta. Sillä on valtava määrä palveluja ja kattavin tietokeskusten välinen verkosto, jonka vuoksi se on saavuttanut suurimman osan suosiostaan. AWS:n selvästi huonoin puoli on sen monimutkainen hinnoittelu palveluilleen. AWS:n IoT Coren hinnoittelu perustuu neljään eri osa-alueeseen; yhteyden kestoon, viestien lukumäärään, varjolaitteisiin ja sääntömoottoriin. Lisäksi maantieteellinen sijainti vaikuttaa näiden osa-alueiden hintoihin. [10-12]

Microsoft Azure on toiseksi suosituin alusta ja yksi suurimmista syistä siihen on yritykset, jotka käyttävät jo valmiiksi muita Microsoftin ohjelmistoja. Azure toimii hyvin yhteen muiden Microsoftin ohjelmistojen kanssa, joten yrityksille on loogista käyttää myös heidän pilvipalvelualustansa. Se tarjoaa monenlaisia palveluja sisältäen muun muassa datan laskennan, analytiikan, varastoinnin ja virtuaalikoneet. Azure IoT Hubin hinnoittelu perustuu viestien lukumäärään päivässä ja valittavissa on joko Basic tai Standard -taso. Basic on halvempi vaihtoehto, mutta siitä puuttuu joitain ominaisuuksia kuten tuki viesteille pilvipalvelusta laitteeseen. Standard-taso kattaa kaikki ominaisuudet, mutta on myös tämän vuoksi kalliimpi. Näiden lisäksi tasot jaetaan vielä eri kokoluokkiin, jotka määräävät lopullisen hinnan. Standard tasosta on olemassa ilmainen luokka, joka kattaa 8000 viestiä päivässä, missä yhden viestin kooksi on määritetty 0,5 KB (kilobyte). Tällöin jos palveluun lähetetään esimerkiksi 0,7 KB:n viesti, se lasketaan kahdeksi viestiksi. Maksulliset luokat lisäävät sekä viestien päivittäistä kokonaismäärää että niille määritettyä yksittäistä kokoa. Esimerkiksi S1, joka on pienin Standard-luokka heti ilmaisen luokan jälkeen, maksaa 25 \$/kk. Se kattaa 400 000 viestiä päivässä ja viestin määritetty koko on 4 KB. Suurimmalla S3-luokalla viestin määritetty koko on sama, mutta päivittäisten viestien sallittu lukumäärä nousee kolmeen miljoonaan asti ja tällöin hinnaksi muodostuu 2500 \$/kk. [10, 11, 13, 14]

Google Cloud on kolmanneksi suosituin pilvipalvelualusta erikoistumalla massadatan laskentaan, analytiikkaan ja koneoppimiseen. Se itse mainostaa alustan tietoturvaa huippuluokkaiseksi ja käyttöä helpoksi. Googlen alustan heikkoutena on palvelujen pieni määrä verrattuna suosituimpiin alustoihin. Google IoT Coren hinnoittelu perustuu datan määrään kuukaudessa. Hinta laskee portaittain, jolloin hinta per MB (megabyte) on suurempi pienemmillä datamäärillä. Ensimmäiset 250 MB ovat

kuitenkin maksuttomia, jonka jälkeen hinta on 250 gigabyteen saakka 0,0045 \$/MB ja lopuksi 5 terabyten jälkeen 0,00045 \$/MB. [10, 11, 15]

Lopuksi esitellään alustojen hintavertailu kolmella eri skenaariolla suuntaa antavasti. Arvioidut hinnat on laskettu alustojen omien laskureiden avulla. AWS:n monimutkaisen hinnoittelun vuoksi laskurissa käytettiin vain laitteiden ja viestien määrää sekä viestien kokoa. Laskurissa on jätetty huomioimatta AWS IoT Coren varjolaitteet ja sääntömoottori. Lisäksi oletettiin, että laitteet ovat yhteydessä alustan IoT Coreen 100% ajasta. Microsoft Azure IoT Hubin hinta laskettiin Standard-tasolla ja valuuttana kaikille alustoille käytettiin US dollareita sekä sijaintina Eurooppaa. Taulukosta 1 nähdään laskureissa käytetyt arvot skenaarioissa, joissa pyrittiin erilaisiin lähestymistapoihin. Ensimmäinen skenaario on sekoitus kahdesta jälkimmäisestä, joissa toisessa laitteiden määrä on vähäinen, yhden viestin koko on suuri ja viestien määrä minuutissa on pieni. Kolmannessa skenaariossa laitteiden määrä ja viestien määrä minuutissa on suuri, mutta yksittäisen viestin koko on pieni. Taulukosta 2 nähdään, että AWS:n IoT Coren käyttö tulee esitellyistä palveluista halvimmaksi kahdessa ensimmäisessä skenaariossa. Microsoft Azure IoT Hub on halvin kolmannessa skenaariossa, jolloin Google Cloud IoT Core jää kalleimmaksi vaihtoehdoksi jokaisessa tapauksessa. Google Cloud IoT Coren hinnoittelu on kuitenkin palveluista selkein.

Taulukko 1. Pilvipalveluiden hintavertailuskenaariot.

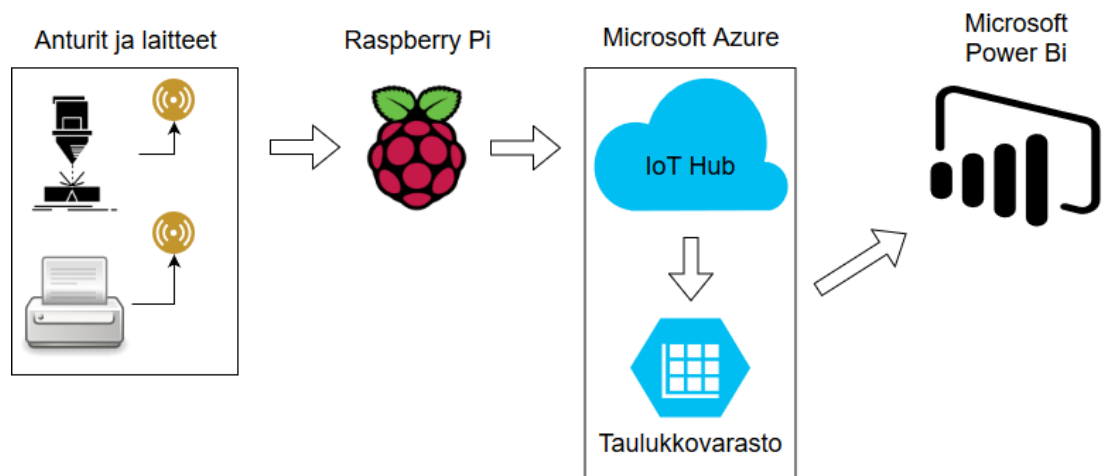
Skenaario	Laitteiden määrä	Viestin koko	Viestien määrä minuutissa	Viestin kokonaismäärä kuukaudessa	Datan määrä kuukaudessa
1.	1000	8 KB	2	8 640 000	691 200 MB
2.	200	50 KB	2	1 728 000	864 000 MB
3.	1000	4 KB	60	2 592 000 000	10 368 000 MB

Taulukko 2. Arvioidut hinnat palveluille eri skenaarioissa.

Skenaario	AWS IoT Core	Azure IoT Hub	Google Cloud IoT Core
1.	212 \$/kk	250,00 \$/kk (S2)	2021,28 \$/kk
2.	208 \$/kk	500,00 \$/kk (2 x S2)	2366,88 \$/kk
3.	2732 \$/kk	2500,00 \$/kk (S3)	13430,94 \$/kk

3. ANTURIDATAN KERÄYS, KÄSITTELY JA VISUALISOINTI

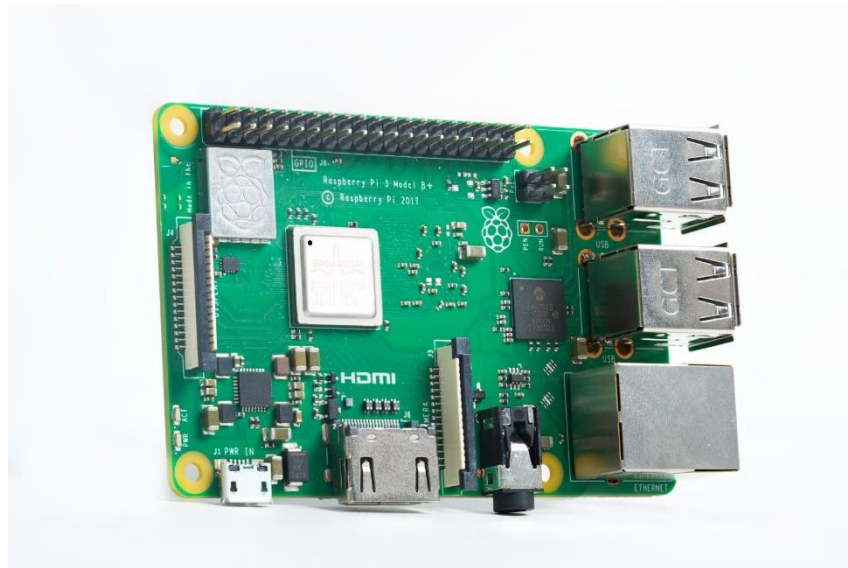
Työn tavoitteena oli kehittää ratkaisua Oulun yliopiston Fab Labin laitteiden käyttöhistorian tarkastelemiseksi. Kuvasta 2 nähdään tämän työn eteneminen ja sen olennaisimmat vaiheet. Data simuloitiin ja lähetettiin Raspberry Pi 3 Model B+ -minitietokoneella Microsoft Azure -pilvipalveluun, jossa se tallennettiin taulukkovarastoon. Lopuksi data voitiin hakea tästä varastosta ja visualisoida Microsoft Power BI:llä. Työssä jätettiin huomioimatta Fab Labin oikeat laitteet, sillä se olisi laajentanut työtä liikaa, eikä se olisi ollut tämän työn kannalta oleellista.



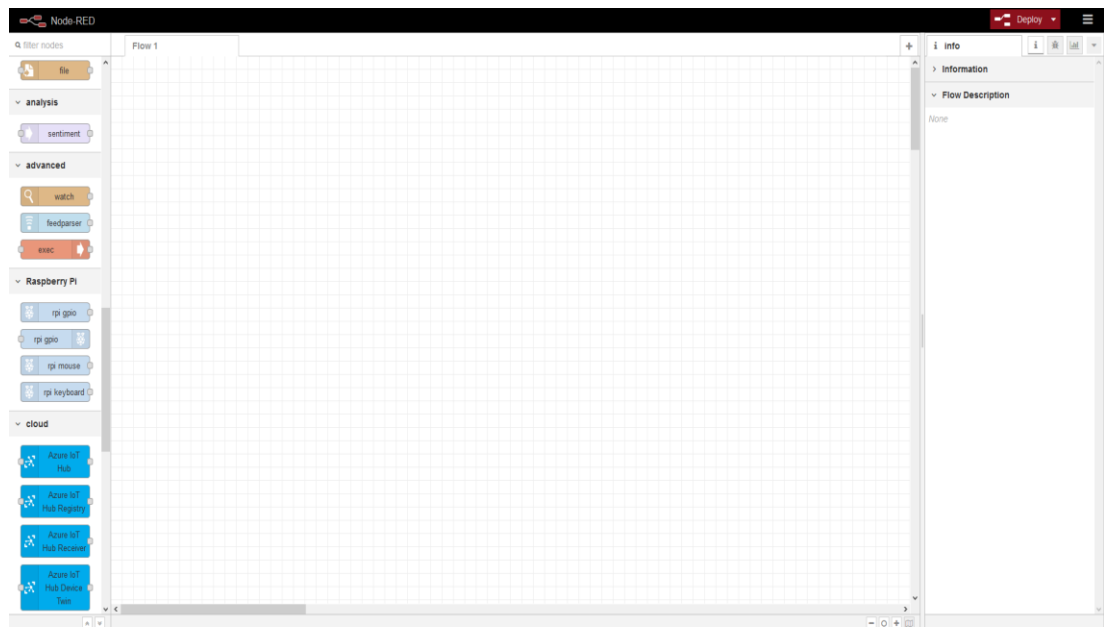
Kuva 2. Työn eteneminen ja sen olennaisimmat vaiheet. Laitteiden anturidata simuloitiin Raspberry Pi:llä.

3.1. Raspberry Pi ja Node-RED

Työosuus aloitettiin valmisteluvaiheella asentamalla kuvassa 3 näkyvään Raspberry Pi -minitietokoneeseen Linux-pohjainen Rasbian käyttöjärjestelmä, jonka jälkeen tietokone yhdistettiin verkkoon. Tämän jälkeen ladattiin ja asennettiin Node-RED työkalu käyttöjärjestelmälle. Node-RED on helppokäyttöinen ja aloittelijaystävällinen JavaScriptiin perustuva ohjelmointityökalu, jossa on selainpohjainen graafinen käyttöliittymä. Kuvassa 4 nähdään Node-RED:in työtila ja sen valmiita eri käyttöön tarkoitettuja toiminnallisia lohkoja, joista saadaan halutun ohjelman vuoesitys yhdistelemällä lohkoja toisiinsa. Jokaiselle lohkolle on määritetty oma tehtävänsä; lohkolle tulee dataa, se tekee saapuneelle datalle jotain ja tämän jälkeen lähettää datan eteenpäin seuraavalle lohkolle [16]. Ohjelman käynnistys tapahtuu joko selaimesta käsin tai suoraan komentoriviltä, jolloin ohjelma suoritetaan Node.JS:llä toteutetussa ympäristössä. Valmisteluvaiheen lopuksi Node-RED:in kirjastoon lisättiin vielä Azure IoT Hub, jonka seurauksena Node-RED:in lohkovalikkoon ilmaantuivat Azure IoT Hub -lohkot.



Kuva 4. Raspberry Pi 3 Model B+. Yhden piirilevyn tietokone. [9]



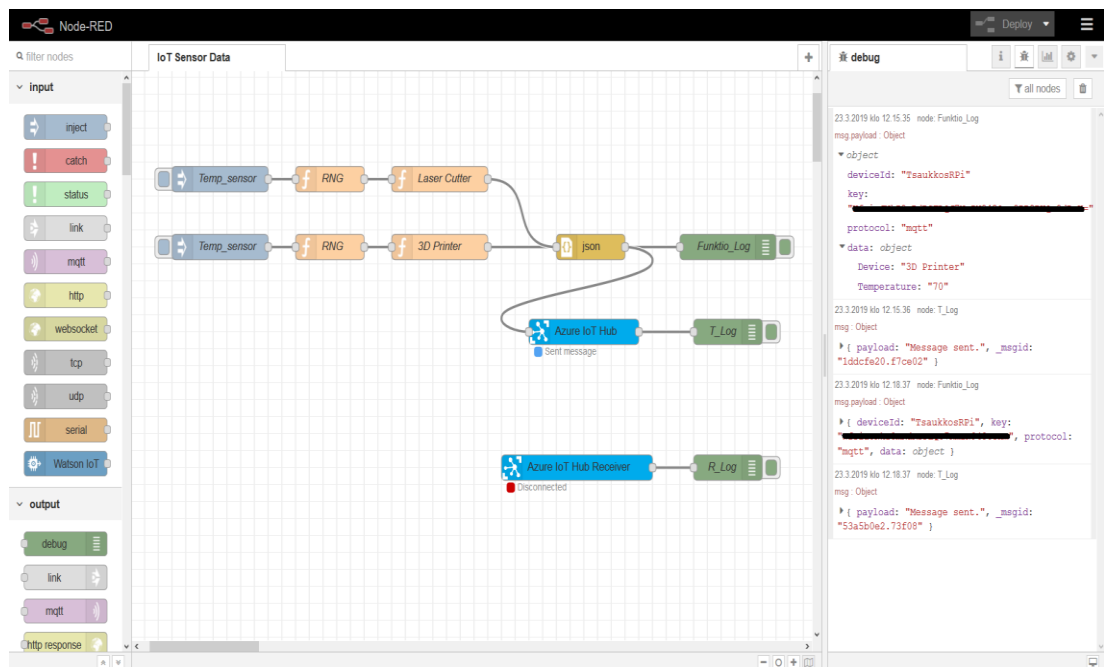
Kuva 3. Node-RED ympäristö ja Azure IoT Hub -lohkot.

Vuoesityksiä voidaan muodostaa vetämällä lohkoja valikosta työtilaan ja yhdistämällä näitä toisiinsa. Tässä työssä käytetyt lohkot ovat:

- Inject, joka toimii datan syöttölohkona. Sitä voidaan käyttää manuaalisesti nappia painamalla tai se voidaan ajastaa syöttämään lähete esimerkiksi sekunnin välein.
- Function, johon kirjoitetaan nimensä mukaisesti jokin funktio, jonka se toteuttaa.
- Json, joka muuttaa merkkijonon json-muotoon tai päinvastoin.

- Azure IoT Hub -lohkot mahdollistavat yhteyden Raspberry Pi:n ja IoT Hubin välillä.
- Debug, joka on systeemin debugausta varten eli sillä nähdään mitä systeemi antaa debug-lohkon kohdalla ulos.

Työssä oleellista oli saada ratkaisu toimimaan useammalla kytketyllä laitteella. Työ toteutettiin aluksi simuloimalla vain yhtä laitetta, jonka jälkeen lisättiin toinen laite alkuperäisen rinnalle. Kuvasta 5 nähdään, että lopulliseen ratkaisuun kuuluivat tämän vuoksi kaksi inject-lohkoa, jotka simuloivat tilannetta, kun laitteiden anturit alkavat lähettää dataa. Ratkaisussa injectit toimivat manuaalisesti nappia painamalla, mutta oikeita antureita käytettäessä inject-lohkot korvattaisiin mqtt-lohkoilla. Injectillä siis lähetetään tässä tapauksessa RNG-lohkolle käsky suorittaa funktio. Tässä esityksessä RNG on funktiolohko, jonka tehtävä on tuottaa vain satunnainen luku ja tätäkään ei todellisessa toteutuksessa käytettäisi. Sen tarkoitus on simuloida lämpötila-arvoja ja lähettää niitä seuraavalle funktiolle. Tämän jälkeen 3D Printer - ja Laser Cutter -lohkot ottavat lämpötilat vastaan ja lisäävät siihen reunalaitteen eli työssä käytetyn Raspberry Pi:n nimen, DeviceId:n eli anturilla mitattavan laitteen nimen, sekä vielä IoT Hubin vaatimat avaimet. Tämän jälkeen json-lohko muuttaa edellä mainitun datan json-muotoon, jolloin Azure IoT Hub-lohko voi ottaa datan vastaan ja lähettää sen IoT Hubille. Azure IoT Hub Receiver -lohkon tehtävä on näyttää Rasbianin terminaalissa tieto datan vastaanottamisesta IoT Hubilla, mutta sekään ei ole systeemin toiminnan kannalta välttämätön. Debug-lohkoja on sijoitettu eri vaihesiin seuraamaan datan kulkua ja auttamaan mahdollisten ongelmien selvityksessä. Kuvan 5 oikeassa reunassa nähdään myös järjestelmän debug-viestit, kun dataa lähetetään IoT Hubille. Valmiin vuoesityksen export-koodi on esitetty liitteessä 1.



Kuva 5. Vuoesitys simuloidun anturidatan lähettämiseksi Azure IoT Hubille käyttäen Node-RED:ia.

3.2. Microsoft Azure

Microsoft Azure tarjoaa opiskelijoille Azure-tilin, jonka luominen on ilmaista, eikä siihen tarvitse luottokorttitietoja. Tiliä käytettäessä on monissa resursseissa valittavissa free tier -taso, jolloin sen käyttö on ilmaista tiettyyn pisteeseen saakka. Tämän lisäksi Microsoft antaa 100 dollarin edestä krediittejä, joita voidaan käyttää alustan maksullisiin resursseihin. Tili on voimassa koko opinto-oikeuden ajan, mutta se täytyy uusia 12 kuukauden välein. [17]

Azure-alustassa luotiin ensimmäisenä IoT Hub, jonka jälkeen työssä käytetty Raspberry Pi täytyi lisätä Hubin IoT-laitteeksi. Tämä onnistui valitsemalla Hubin IoT-laitteet ja lisäämällä sinne uuden laitteen. Laitteelle täytyi antaa nimi, jonka jälkeen IoT Hub tuotti sille automaattisesti yksilöllisen tunniste. IoT Hubin oma yksilöllinen tunniste, jonka alkuosana toimii sen isäntänimi, löydetään Hubin asetuksista. Laitteen yhteysmerkkijonoa ja siihen sisältyvää avainosaa, sekä IoT Hubin yhteysmerkkijonoa ja sen isäntänimeä tarvitaan Node-RED'in vuoesityksen lohkoissa, jotta data saadaan siirtymään oikealle Hubille.

Oletusarvoisesti Azuren IoT Hubilta data siirtyy valmiiseen päätepisteeseen, jossa se säilyy maksimissaan 7 vuorokautta [18]. Ratkaisun kannalta oleellinen seikka oli datan varastointi, jotta sitä voidaan tarkastella myöhemmin. Tämän vuoksi Azuressa täytyi tehdä erillinen varastotili. Varastotilille on mahdollista tehdä neljä erilaista tarkoitukseen soveltuvaa datavarastoa [19]:

- Azure Blobs on optimoitu valtavan jäsentämättömän datamäärän varastointiin. Tällaista voi olla esimerkiksi teksti- tai binääridata.
- Azure Filesä käytetään tiedostojen jakamiseen. URL:iin ja siihen liitetyn tunnuksen avulla voidaan jakaa tiedosto muille.
- Azure Queues on pienikokoisten jonomuotoisten viestien tallentamista ja noutamista varten.
- Azure Table on optimoitu suurien jäsenettyjen datamäärien varastointiin.

Tämän ratkaisun tarkoitukseen luontevin oli table storage eli taulukkovarasto, sillä data oli jäsenelty etukäteen. Tämän vuoksi tilille tehtiin taulukkovarasto, mutta datan tallentaminen IoT Hubilta suoraan tähän varastoon ei ollut mahdollista. Taulukkovarastoon tallentamiseen jouduttiin käyttämään Azuren toista resurssia, Function App:iä.

Function App on tehty pienten koodien käyttämiseen alustan sisällä. Funktiot voidaan kirjoittaa muun muassa C#, F#, Node.js, Java, tai PHP -kieliä käyttäen. Azuren opiskelijatilillä Function App'in käyttö on ilmaista miljoonaan kuukaussittaiseen ajoon asti, mutta tämän jälkeen sen käyttäminen maksaa. Valittavissa on kaksi vaihtoehtoa, jolla hinta määräytyy: kulutuksen eli funktion ajokertojen perusteella tai kuukausimaksulla yhdessä muiden Azuren palveluiden kanssa. [20]

Seuraavaksi Azureen tehtiin funktio käyttäen C-kieltä (liite 2), jonka tehtävä oli tallentaa Raspberry Pi:ltä tuleva data taulukkoon. Funktio käynnistyy, kun IoT Hubille tulee viesti, jolloin se luo taulukkoon rivin, jossa on päivänmäärä ja kellonaika. Tämän jälkeen funktio ottaa viestistä laitteen nimen sekä lämpötila-arvon ja lisää ne lopuksi luodulle riville. Taulukon sisältöä voidaan tarkastella kuvassa 6 esiintyvällä Storage Explorer -työkalulla. Työkalu löytyy Azure-alustan varastotililtä tai se voidaan ladata työpöytäversioksi tietokoneelle, jolloin kirjautumalla Azure tunnuksilla nähdään

varastoitu data. Kuvassa 7 näkyy Azure-alustan Dashboard, josta voidaan nähdä lopullisessa ratkaisussa käytetyt resurssit. Käytettyjen Azure resurssien konfigurointien ruutukaappauksia on esitetty liitteessä 3.

Dashboard > dataa - Storage Explorer (preview)

dataa - Storage Explorer (preview)

Search (Ctrl+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Advanced Threat Protection

Properties

Locks

Export template

Blob service

Blobs

Custom domain

Soft delete

Azure CDN

Add Azure Search

BLOB CONTAINERS

FILE SHARES

QUEUES

TABLES

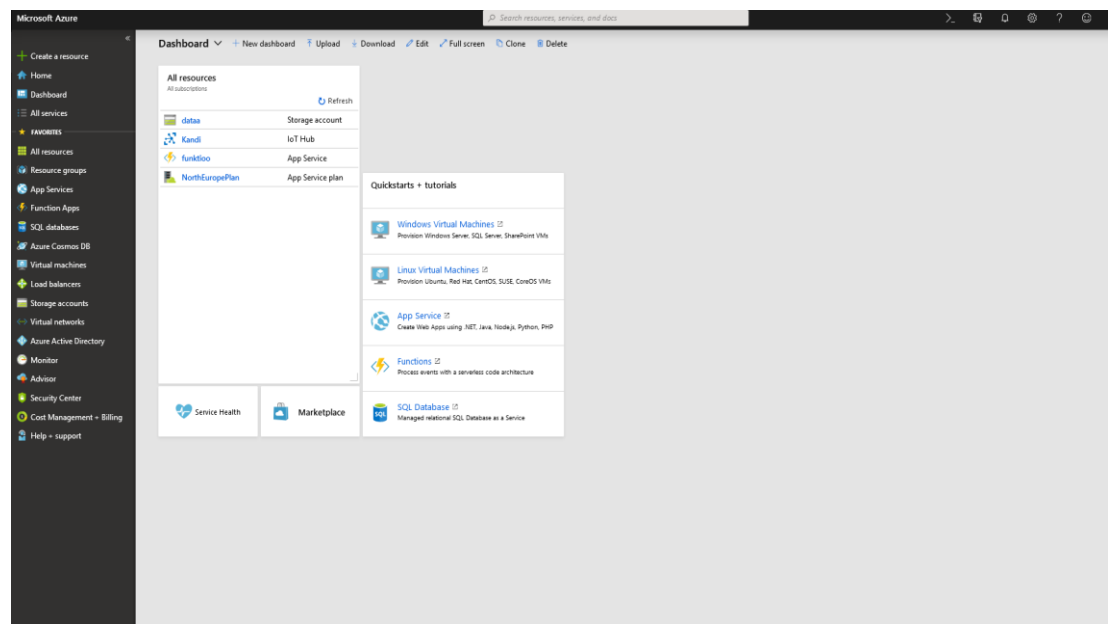
Anturit

Query + Add Edit Select All Column Options Delete Table Statistics Refresh

PARTITIONKEY	ROWKEY	TIMESTAMP	DEVICE	TEMPERATURE
2018	17-11-2018 16:14:39	2019-02-12T14:21:32.255614Z	Laser Cutter	15
2018	17-11-2018 16:14:44	2019-02-12T14:20:52.9828694Z	Laser Cutter	95
2018	17-11-2018 16:14:46	2019-02-12T14:21:28.9352713Z	Laser Cutter	97
2018	17-11-2018 16:14:53	2019-02-12T14:21:25.2756837Z	Laser Cutter	50
2018	17-11-2018 16:14:54	2018-11-17T14:14:54.0385087Z	3D Printer	47
2018	17-11-2018 16:14:55	2018-11-17T14:14:55.8848145Z	3D Printer	12
2018	17-11-2018 16:14:56	2019-02-12T14:21:21.8232478Z	Laser Cutter	94
2018	17-11-2018 16:14:57	2018-11-17T14:14:57.6140401Z	3D Printer	62
2018	17-11-2018 16:14:58	2019-02-12T14:21:18.0926093Z	Laser Cutter	90
2018	17-11-2018 16:14:59	2018-11-17T14:14:59.5263958Z	3D Printer	83
2018	17-11-2018 16:15:01	2018-11-17T14:15:01.2706312Z	3D Printer	90
2018	17-11-2018 16:15:02	2019-02-12T14:21:14.1618303Z	Laser Cutter	99
2018	17-11-2018 16:15:59	2019-02-12T14:21:10.3521424Z	Laser Cutter	18
2018	17-11-2018 16:16:00	2018-11-17T14:16:00.479581Z	3D Printer	25
2018	17-11-2018 16:16:01	2018-11-17T14:16:01.3301814Z	3D Printer	38
2018	17-11-2018 16:16:02	2019-02-12T14:21:04.5010077Z	Laser Cutter	31
2018	17-11-2018 16:16:27	2018-11-17T14:16:27.2755631Z	3D Printer	12
2018	17-11-2018 16:17:01	2019-02-12T14:21:00.0188392Z	Laser Cutter	36
2018	17-11-2018 16:17:03	2019-02-12T14:20:56.5383834Z	Laser Cutter	32
2018	17-11-2018 16:17:06	2018-11-17T14:17:06.6047777Z	3D Printer	55
2018	17-11-2018 16:17:45	2019-02-12T14:20:49.3292888Z	Laser Cutter	6
2018	17-11-2018 16:17:46	2018-11-17T14:17:46.3999737Z	3D Printer	86
2018	17-11-2018 16:45:35	2019-02-12T14:20:45.7137342Z	Laser Cutter	7
2018	17-11-2018 16:46:43	2018-11-17T14:46:43.6742632Z	3D Printer	49
2018	17-11-2018 16:46:45	2019-02-12T14:20:40.0147073Z	Laser Cutter	18
2018	17-11-2018 17:01:29	2019-02-12T14:20:36.127959Z	Laser Cutter	77
2018	17-11-2018 17:01:31	2019-02-12T14:20:31.6768184Z	Laser Cutter	80
2018	17-11-2018 17:06:38	2018-11-17T15:06:38.6231093Z	3D Printer	68
2018	17-11-2018 17:12:54	2019-02-12T14:20:27.5288858Z	Laser Cutter	42
2018	17-11-2018 17:12:55	2019-02-12T14:20:23.5300585Z	Laser Cutter	28
2018	17-11-2018 17:12:56	2019-02-12T14:20:19.3681231Z	Laser Cutter	91
2018	17-11-2018 17:29:47	2019-02-12T14:20:15.2232252Z	Laser Cutter	34
2019	12-02-2019 15:09:35	2019-02-12T14:20:08.425473Z	Laser Cutter	50
2019	12-02-2019 15:30:20	2019-02-12T13:30:20.5997772Z	3D Printer	95

Showing 1 to 81 of 81 cached items

Kuva 6. Azuren sisäinen Storage Explorer, jolla voidaan katsoa taulukon sisältöä.



Kuva 7. Azure Dashboard. Lopullisessa ratkaisussa käytetyt alustan resurssit.

3.3. Microsoft Power BI

Microsoft Power BI Desktop on työkalu datan visualisointiin. Haluttu data voidaan hakea niin paikallisesta kuin pilvipohjaisestakin lähteestä ja sen tukemia lähteitä ovat muun muassa Dynamics 365, Salesforce, Azure SQL DB (Structured Query Language Database), Excel ja SharePoint. Lähteestä haettu data voidaan päivittää manuaalisesti tai ohjelma voidaan määrittää päivittämään se automaattisesti. Power BI Desktopin ominaisuuksiin kuuluvat esimerkiksi datalle tehtävät nopeat laskelmat, ennusteet sekä sen yhdistelyt ja ryhmittelyt. Se tukee Excelin tapaan myös DAX (Data Analysis Expressions)-kyselykieltä, jolla voidaan tehdä monimutkaisiakin ratkaisuja esitykseen. Datan visualisointiin ohjelmasta löytyy valmiiksi erilaisia taulukoita, diagrammeja ja kaavioita. Tämän lisäksi niitä voidaan ladata lisää sekä Microsoftin että muiden käyttäjien tekemänä Microsoftin omalta kauppapaikalta. [21]

Ohjelman työpöytäversiolla tehty esitys voidaan julkaista Power BI services - palveluun, jolloin esityksen näkee myös omasta työtilasta kirjautumalla Power BI:n verkkosivuille. Palveluun on kahden tyyppisiä lisenssejä: käyttäjäkohtaisia ja kapasiteettipohjaisia. Ensimmäisenä mainitut jakautuvat vielä kahteen tyyppiin:

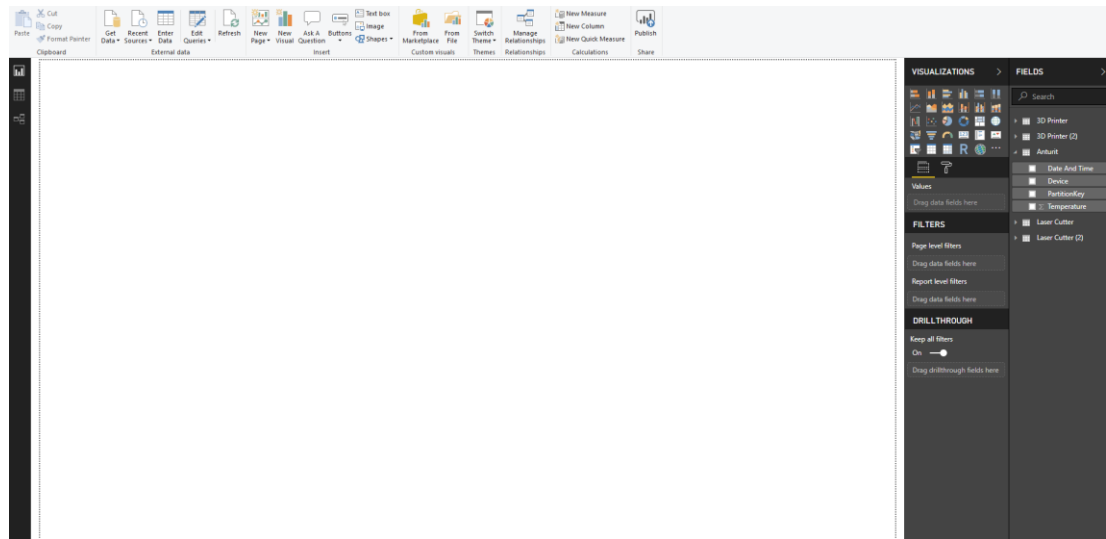
- Ilmaiseen lisenssiin, joka kattaa yli 70 datalähteen käytön, esityksen julkaisemisen verkkoon ja esityksen viennin PowerPointiin, Exceliin ja CSV-tiedostomuotoon.
- Pro lisenssiin, joka maksaa 8,40 € kuukaudessa. Siihen kuuluu ilmaisen lisenssin ominaisuudet ja lisäksi mahdollisuus esityksen jakamiseen sähköpostilla muille Pro lisenssin omistajille. Muita ominaisuuksia ovat muun muassa App -työtilat ja analysointi Excelissä ja Power BI Desktopissa.

Kapasiteettipohjaiset eli premium lisenssit ovat tarkoitettu suuremmille ryhmille, kuten yrityksille. Laskenta suoritetaan Microsoftin hallinnoimilla laitteilla, jonka ansioista saavutetaan käyttäjäkohtaista lisenssiä parempi suorituskyky ja suurempi datamäärä. Sisällön tuottamiseen tarvitaan kuitenkin myös Pro lisenssi, mutta sisältöä voidaan jakaa laajasti ilman, että vastaanottajalla täytyy olla tätä lisenssiä. Premium lisenssin hinta määräytyy tarvittavan kapasiteetin perusteella. Esimerkiksi Power BI Premium laskurilla laskettuna arvioiduksi hinnaksi 100 käyttäjälle, joista 10 on Pro lisensoituja, tulee 4212,30 € kuukaudessa. [21, 22]

Ratkaisun lopullinen datan esitys tapahtui Power BI Desktopilla. Laitteiden lämpötilat haluttiin saada näkyviin päivänmäärän ja kellonajan funktiona. Lisäksi haluttiin nähdä laitteiden päälläoloaika, joka vaati hieman soveltamista yksittäisten simuloitujen arvojen vuoksi. Todellinen toteutus toimisi seuraavalla periaatteella: laite lähettää arvon 1, kun virran kulutus nousee tietyn rajan yli ja arvon 0, kun virran kulutus laskee tämän rajan alle. Tämän jälkeen voitaisiin laskea näiden arvojen aikaleimojen erotus. Esityksessä tämä toteutettiin niin, että kahden peräkkäisen aikaleiman erotuksen ollessa alle 10 sekuntia arvo hyväksytään ja laite oletetaan päällä olevaksi. Arvon ollessa suurempi se jätetään huomioimatta.

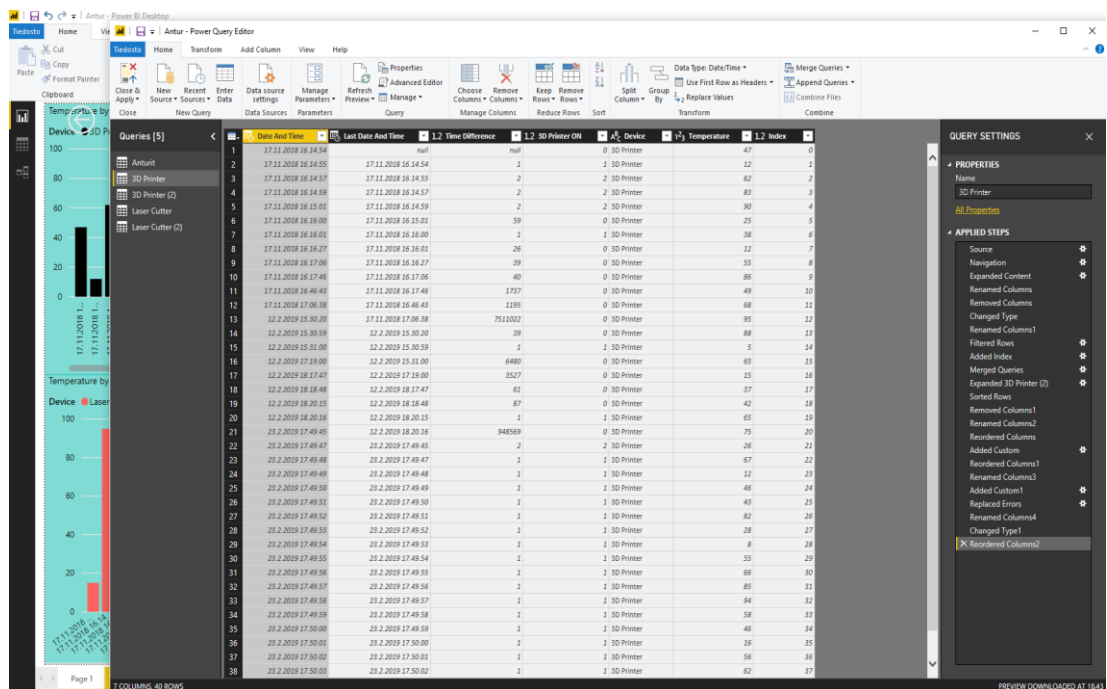
Kuvassa 8 nähdään Power BI:n aloitusnäky. Keskellä on työtila, jossa erilaiset taulukot ja kuvaajat voidaan esittää valitsemalla haluttu data ja esitystapa reunan valikoista. Datalle voidaan tehdä visuaalisia suodatuksia ja yksinkertaisia laskutoimituksia käyttämällä valmiita työkaluja. Lisäksi voidaan käyttää DAX-

kyselykieltä esimerkiksi monimutkaisemmille ja pidemmille laskutoimituksille. Tehtäväpalkista löytyy työkalut myös datatyyppin ja sen formaatin valintaan, mikä on hyödyllistä esimerkiksi päivänmäärien kanssa työskennellessä.



Kuva 8. Power BI aloitusnäkö ja visuaalisen esityksen työtila.

Visualisoinnin luominen aloitettiin hakemalla data Azuren varastotililtä syöttämällä sen osoite Power BI:lle. Tämän jälkeen avautui Query editor -näkö, josta taulukon laajentamalla saatiin data näkyviin. Query editorissa voidaan suodattaa ja muotoilla dataa, jotta se on helpompi esittää visuaalisesti. Seuraavaksi editorilla suodatettiin taulukosta epäolennainen data pois, jonka jälkeen tehtiin tarvittavat vähennyslaskut ja muut operaatiot 3D-tulostimen päälläoloajan selvittämiseksi. Tämän jälkeen päälläoloaika lisättiin omaan 3D Printer ON -sarakeeseensa kuvan 9 mukaisesti, jossa arvot esiintyvät sekunteina. Samat vaiheet toistettiin vielä toiselle laitteelle, jonka jälkeen muutokset tallennettiin ja hyväksyttiin.

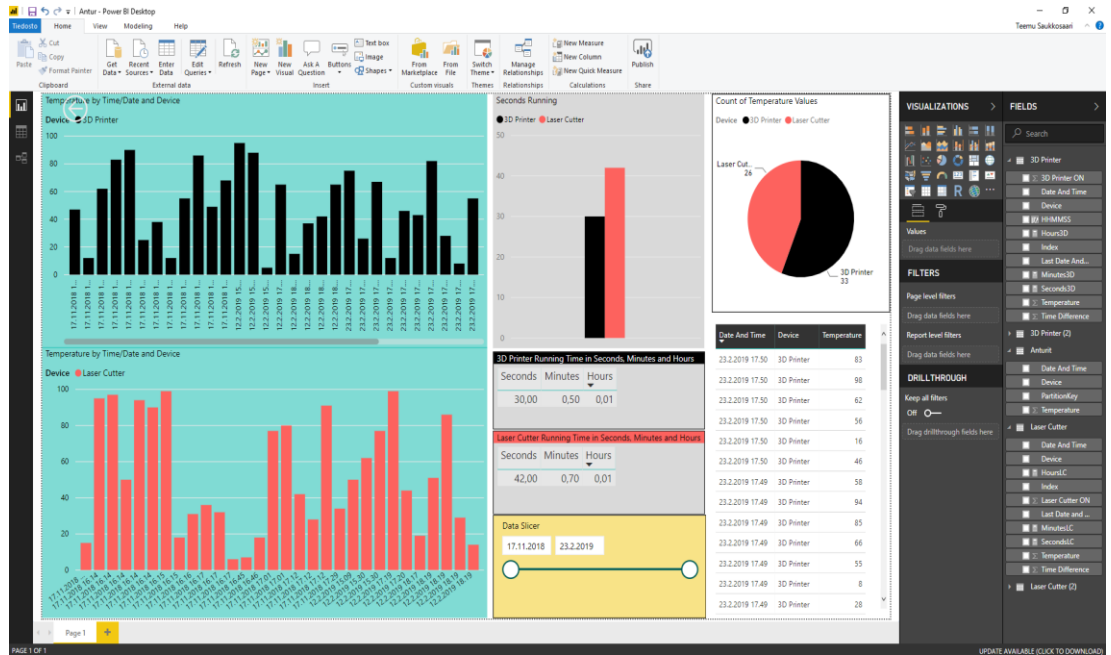


Kuva 9. Power BI Query Editor -näkömä. 3D-tulostimen data.

Lopullinen datan visuaalinen esitys tapahtui Power BI:n aloitusnäkymässä. Esityksen visualisoinnin valmistelu aloitettiin laajentamalla datan taulukot, jolloin saatiin valittua halutut sarakkeet. Laitteiden päälläoloaika varten summattiin tarvittavien sarakkeiden sekunnit yhteen käyttäen DAX-kyselykieltä. Työn kohdetta ajatellen oli myös oleellista saada muutettua sekunnit minuuteiksi ja tunneiksi, joten laskutoimitukset tehtiin yksikkömuunnoksillekin. Tämän jälkeen myös tehdyt operaatiot ilmaantuivat reunan valikkoon, jolloin nekin pystyttiin valitsemaan ja visualisoimaan.

Sopivat kuvaajat ja taulukot valittiin eri esityksille sekä tehtiin tarvittavat suodatukset, jotta kuvaajat olisivat mahdollisimman selkeitä ja niistä näkyisi oikea informaatio. Esitykseen lisättiin myös slicer-työkalu, jolla voidaan rajata dataa päivänmäärän tarkkuudella. Ratkaisun kannalta olisi ollut hyvä, jos dataa olisi voinut rajata tarkemmin, mutta Power BI:n valmiilla työkalulla se ei ollut mahdollista. Kuvassa 10 nähdään lopullinen visuaalinen esitys simuloidulle datalle, missä esiintyy:

- Kummallekin laitteelle lämpötilakuvaajat ajan funktiona.
- Taulukot, jossa näkyvät laitteiden päälläoloajat eri aikayksiköissä.
- Laitteiden yhteinen pylväsdiagrammi päälläoloajoista sekunteina.
- Työkalu päivänmäärän rajaamiseen.
- Ympyrädiagrammi lämpötila-arvojen lukumäärän laskemiseen.
- Yhteinen taulukko laitteiden lämpötila-arvoista.



Kuva 8. Lopullinen visuaalinen esitys simuloidusta anturidatasta.

4. YHTEENVETO

Internet of things on laaja käsite, jolle ei ole yhtä selkeää määritelmää. Tässä tutkielmassa on pyritty lyhyesti avaamaan tätä käsitettä, kertomaan sen vaadituista ominaisuuksista ja antamaan esimerkkejä mahdollista sovelluskohteista. Lisäksi tutkielmassa esiteltiin muutamia laitteisto- ja pilvipalvelualustoja, joita voidaan käyttää erilaisissa IoT-järjestelmissä.

Tämän työn lopussa esiteltiin IoT-ratkaisu, jota kehitettiin Oulun yliopiston Fab Labiin. Tavoitteena oli saada tallennettua anturidataa pilvipalveluun, josta se on mahdollista hakea myöhemmin tarkasteltavaksi. Lisäksi ratkaisuun kuului vielä datan visualisointi, jolla saatiin esitettyä tulokset selkeämmin. Anturidata simuloitiin Raspberry Pi -minitietokoneella, pilvipalvelualustana käytettiin Microsoft Azurea ja lopuksi data visualisoitiin Microsoft Power BI:llä.

Tätä työtä hyödyntäen ja jalostaen Oulun yliopiston Fab Lab voisi kehittää ratkaisun kaikkien laitteiden käyttöhistorian tarkasteluun, sekä laajentaa sen myös kattamaan reaaliaikaisen datan tarkastelun.

5. LÄHTEET

- [1] Vodafone (luettu 25.10.2019) Emergency One. URL: <https://www.vodafone.com/business/news-and-insights/case-study/emergency-one>
- [2] Yan Z., Zhang P. & Vasilakos A. (2014) A survey on trust management for Internet of Things. In: Journal of Network and Computer Applications, Vol. 42, s. 120-134.
- [3] Minerva R., Biru A. & Rotondi D. (2015) Towards a Definition of the Internet of Things (IoT). IEEE/Telecom Italia.
- [4] Fuller JR (luettu 4.5.2019) The 4 stages of an IoT architecture. URL: <https://techbeacon.com/enterprise-it/4-stages-iot-architecture>
- [5] Lee I. & Lee K. (2015) The Internet of Things (IoT): Applications, investments, and challenges for enterprises. In: Business Horizons, Vol. 58, s. 431-440.
- [6] Ding N., Wagner D., Xiaomeng C., Pathak A., Hu Y. & Rice A. (2013) Characterizing and modeling the impact of wireless signal strength on smartphone battery drain. In: ACM SIGMETRICS Performance Evaluation Review, Vol. 41, s. 29-40.
- [7] Arduino (luettu 20.4.2019) URL: <https://www.arduino.cc/en/guide/introduction>
- [8] Texas Instruments (luettu 25.10.2019) SensorTag. URL: <http://www.ti.com/tool/TIDC-CC2650STK-SENSORTAG>
- [9] Raspberry Pi Foundation (luettu 17.3.2019) Raspberry Pi 3 Model B plus. URL: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>
- [10] Harvey C. & Patrizio A. (luettu 15.6.2019) AWS vs. Azure vs. Google: Cloud Comparison [2019 Update]. URL: <https://www.datamation.com/cloud-computing/aws-vs-azure-vs-google-cloud-comparison.html>
- [11] Skerrett I. (luettu 15.6.2019) Price Comparison of IoT Platform Vendors. URL: <https://medium.com/@iskerrett/price-comparison-of-iot-platform-vendors-b07ab4bbf0e>
- [12] Amazon (luettu 15.6.2019) AWS IoT Core pricing. URL: <https://aws.amazon.com/iot-core/pricing/>

- [13] Microsoft Azure (luettu 15.6.2019) Azure IoT Hub pricing. URL: <https://azure.microsoft.com/en-us/pricing/details/iot-hub/>
- [14] Patil Y. (2017) Azure IoT Development Cookbook. Pack Publishing Ltd. Birmingham, UK.
- [15] Google (luettu 15.6.2019) Google Cloud IoT Core Pricing. URL: <https://cloud.google.com/iot/pricing>
- [16] JS Foundation (luettu 9.3.2019) Node-RED. URL: <https://nodered.org/>
- [17] Microsoft Azure (luettu 25.3.2019) Azure for Students FAQ. URL: <https://azure.microsoft.com/en-in/free/free-account-students-faq/>
- [18] Microsoft Azure (luettu 23.3.2019) Built-in endpoint. URL: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-messages-read-builtin>
- [19] Microsoft Azure (luettu 24.3.2019) Introduction to Azure Storage. URL: <https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction>
- [20] Microsoft Azure (luettu 26.10.2019) Azure Functions pricing. URL: <https://azure.microsoft.com/en-us/pricing/details/functions/>
- [21] Microsoft Power BI (luettu 25.3.2019) Power BI desktop. URL: <https://powerbi.microsoft.com/en-us/desktop/>
- [22] Microsoft Power BI (luettu 26.3.2019) Power BI. URL: <https://docs.microsoft.com/en-us/power-bi/>

6. LIITTEET

Liite 1	Node-RED export
Liite 2	Azure Function App -koodi
Liite 3	Käytettyjen Azure resurssien konfigurointien ruutukaappauksia


```
[{"id":"742c8931.a25bb8","type":"tab","label":"IoT Sensor Data","disabled":false,"info":"","z":742c8931.a25bb8,"name":"Temp_sensor","topic":"Temperature","payload":"30","payloadType":"num","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":190,"y":300,"wires":[["381d6381.1daef4"]]}, {"id":"342db912.eb6a66","type":"function","z":742c8931.a25bb8,"name":"3D Printer","func":"payload1 = '{\"deviceId\":\"TsaukkosRPi\",';\npayload1 = payload1 + \"key\":\"WfsiaJNkI8mRdRSFLQJ7NmZN943Oon8DR2BWQw9dRoU=\";\npayload1 = payload1 + \"protocol\":\"mqtt\";\npayload1 = payload1 + \"data\":{\"Device\":\"3D Printer\", \"\" + msg.topic + \": \"\" + msg.payload + \"}\"}";\n\nnewpayload = { payload: payload1 }; \nreturn newpayload;","outputs":1,"noerr":0,"x":510,"y":300,"wires":[["59013c91.a44c04"]]}, {"id":"159093b8.6b0d2c","type":"debug","z":742c8931.a25bb8,"name":"Funktio_Log","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","x":930,"y":300,"wires":[]}, {"id":"59013c91.a44c04","type":"json","z":742c8931.a25bb8,"name":"","property":"payload","action":"","pretty":false,"x":730,"y":300,"wires":[["159093b8.6b0d2c","99f773ac.0e97f8"]]}, {"id":"2929f2c9.029b46","type":"debug","z":742c8931.a25bb8,"name":"T_Log","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":true,"x":930,"y":400,"wires":[]}, {"id":"99f773ac.0e97f8","type":"azureiothub","z":742c8931.a25bb8,"name":"Azure IoT Hub","protocol":"mqtt","x":720,"y":400,"wires":[["2929f2c9.029b46"]]}, {"id":"4db705ee.aa9dec","type":"azureiothubreceiver","z":742c8931.a25bb8,"name":"Azure IoT Hub Receiver","x":710,"y":560,"wires":[["7c5cca8b.f37cbc"]]}, {"id":"7c5cca8b.f37cbc","type":"debug","z":742c8931.a25bb8,"name":"R_Log","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":true,"x":930,"y":560,"wires":[]}, {"id":"381d6381.1daef4","type":"function","z":742c8931.a25bb8,"name":"RNG","func":"msg.payload = Math.round(Math.random()*100);\nreturn msg;","outputs":1,"noerr":0,"x":350,"y":300,"wires":[["342db912.eb6a66"]]}, {"id":"f6a4bb00.9ea05","type":"inject","z":742c8931.a25bb8,"name":"Temp_sensor","topic":"Temperature","payload":"300","payloadType":"num","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":190,"y":220,"wires":[["85d8f58e.74b0b"]]}, {"id":"fe7dda4e.c9854","type":"function","z":742c8931.a25bb8,"name":"Laser Cutter","func":"payload1 = '{\"deviceId\":\"TsaukkosRPi\",';\npayload1 = payload1 + \"key\":\"WfsiaJNkI8mRdRSFLQJ7NmZN943O*****\"; \npayload1 = payload1 + \"protocol\":\"mqtt\";\npayload1 = payload1 + \"data\":{\"Device\":\"Laser Cutter\", \"\" + msg.topic + \": \"\" + msg.payload + \"}\"}";\n\nnewpayload = { payload: payload1 }; \nreturn newpayload;","outputs":1,"noerr":0,"x":510,"y":220,"wires":[["59013c91.a44c04"]]}, {"id":"85d8f58e.74b0b","type":"function","z":742c8931.a25bb8,"name":"RNG","func":"msg.payload = Math.round(Math.random()*100);\nreturn msg;","outputs":1,"noerr":0,"x":350,"y":220,"wires":[["fe7dda4e.c9854"]]}]
```

```
#r "Microsoft.WindowsAzure.Storage"

using System;
using Microsoft.WindowsAzure.Storage.Table;

public static void Run(DataPoint myIoTHubMessage, out DataPoint
outputTable, TraceWriter log)
{
    log.Info($"C# IoT Hub trigger function processed a message:
{myIoTHubMessage}");

    DateTime dt = DateTime.Now;
    dt = dt.AddHours(2);

    myIoTHubMessage.PartitionKey = DateTime.Now.Year.ToString();
    myIoTHubMessage.RowKey = dt.ToString("dd-MM-yyyy H:mm:ss");

    outputTable = myIoTHubMessage;
}

public class DataPoint : TableEntity
{
    public string Device { get; set; }
    public double Temperature { get; set; }
}
```

IoT hub - Microsoft Azure x | +

https://portal.azure.com/#create/Microsoft.IoTHub

Microsoft Azure

Search resources, services, and docs (G+/)

Home > New > IoT Hub > IoT hub

IoT hub

Microsoft

Basics Size and scale Review + create

Create an IoT Hub to help you connect, monitor, and manage billions of your IoT assets. [Learn More](#)

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure for Students

Resource Group * ⓘ IoT

Region * ⓘ North Europe

IoT Hub Name * ⓘ Nimhubille

Review + create Next: Size and scale > Automation options

IoT hub - Microsoft Azure x | +

https://portal.azure.com/#create/Microsoft.IoTHub

Microsoft Azure

Search resources, services, and docs (G+/)

Home > New > IoT Hub > IoT hub

IoT hub

Microsoft

Basics Size and scale Review + create

Each IoT Hub is provisioned with a certain number of units in a specific tier. The tier and number of units determine the maximum daily quota of messages that you can send. [Learn more](#)

SCALE TIER AND UNITS

Pricing and scale tier * ⓘ F1: Free tier

Only one Free IoT Hub is allowed per subscription. [Learn how to choose the right IoT Hub tier for your solution](#)

Number of F1 IoT Hub units ⓘ 1

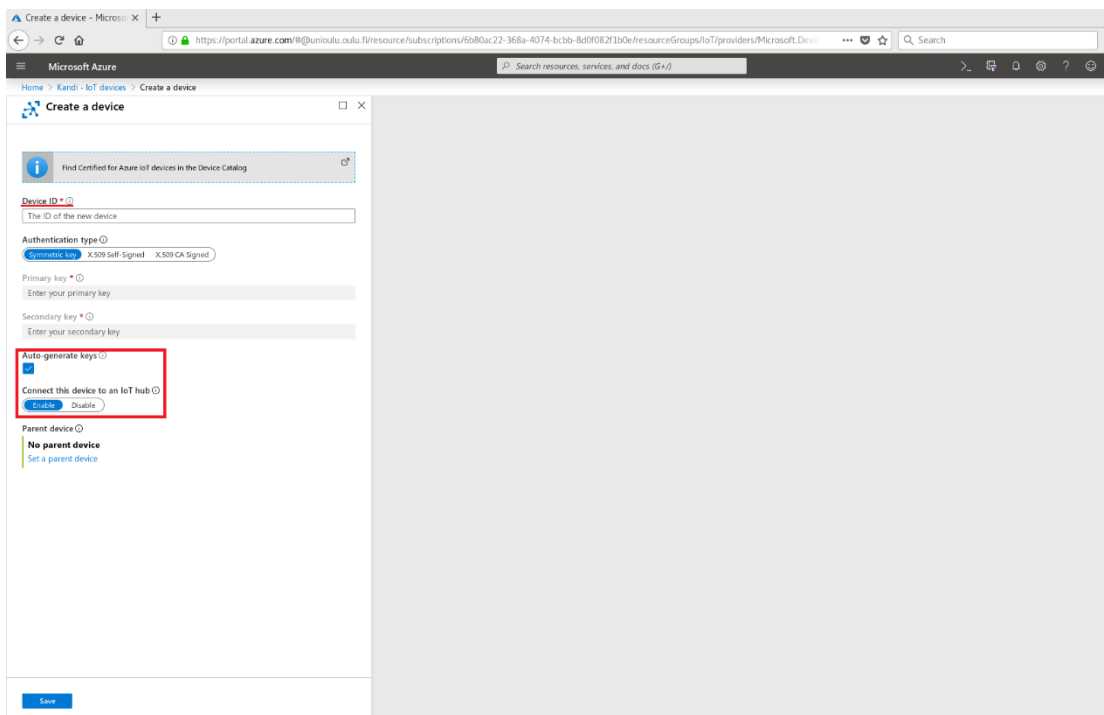
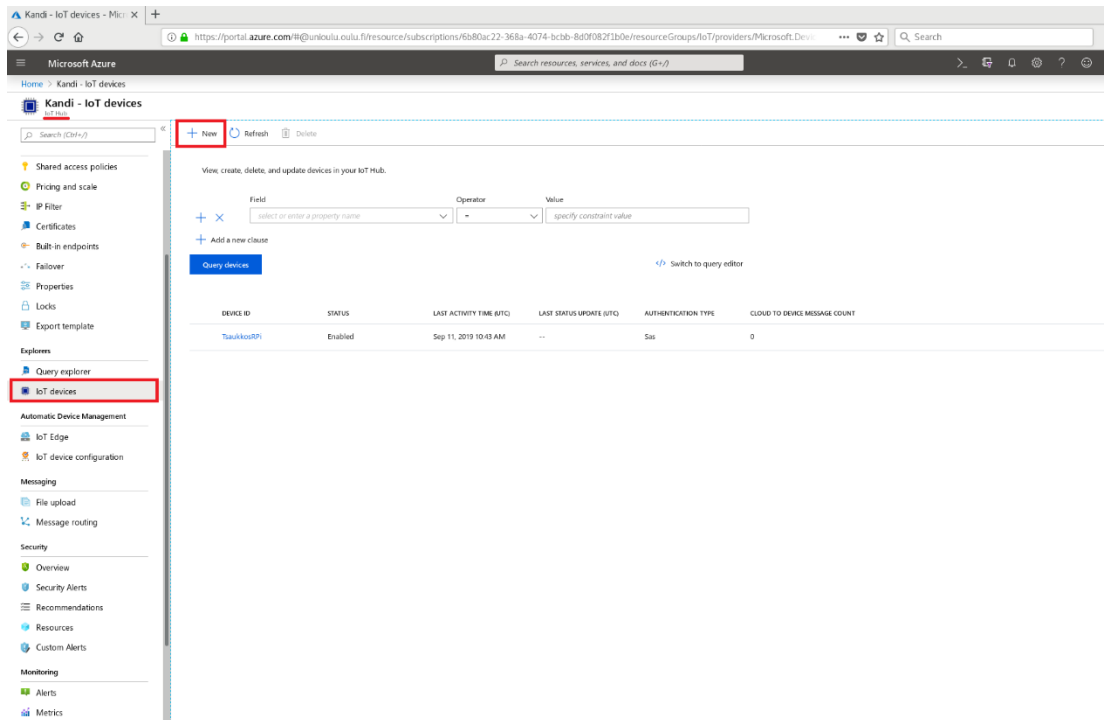
This determines your IoT Hub scale capability and can be changed as your need increases.

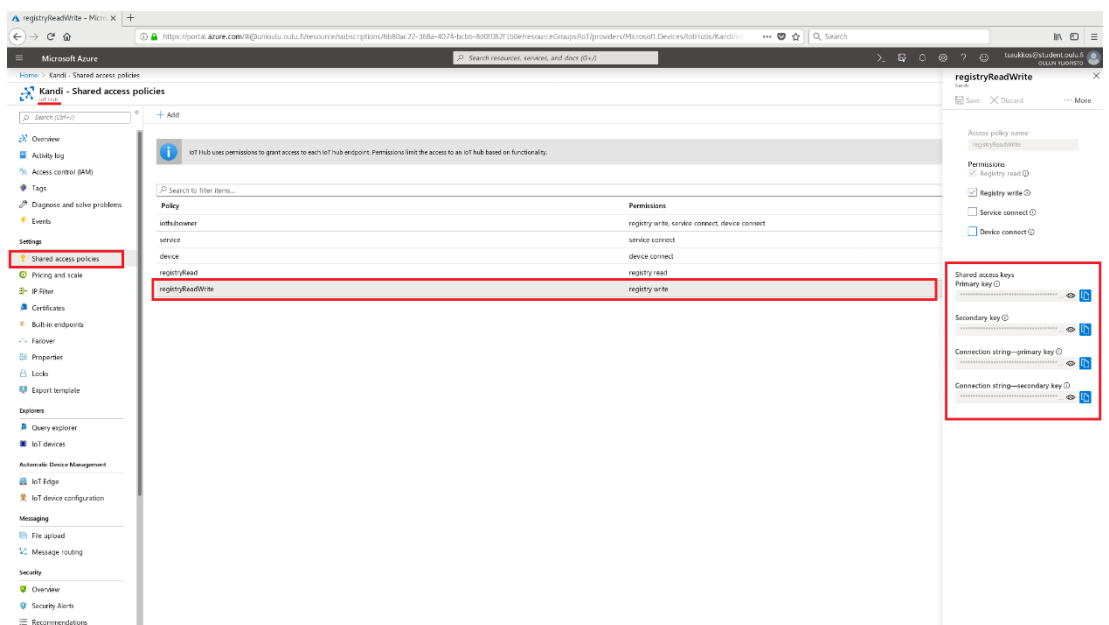
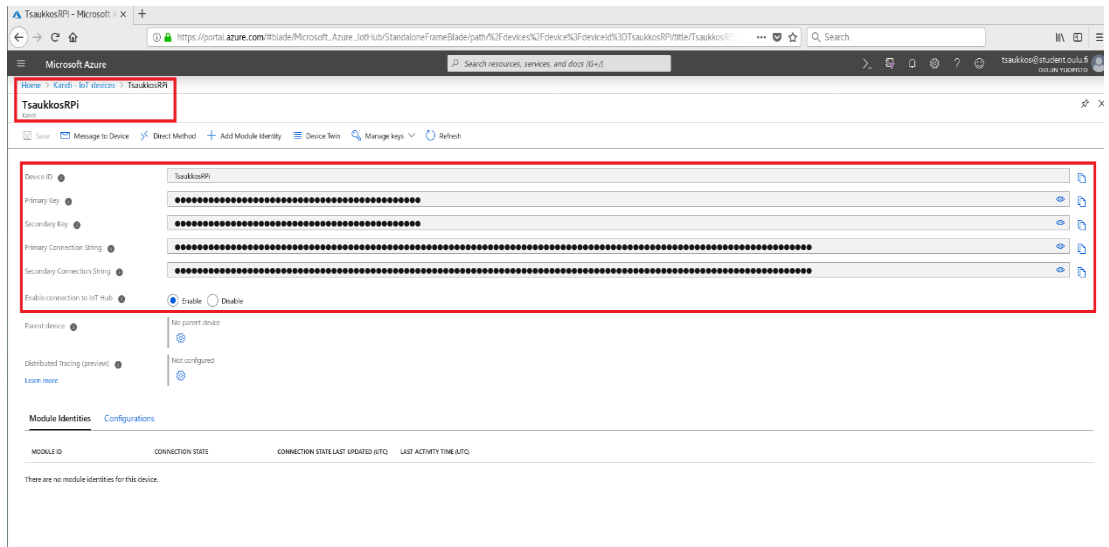
Pricing and scale tier ⓘ F1	Device-to-cloud messages ⓘ Enabled
Messages per day ⓘ 8,000	Message routing ⓘ Enabled
Cost per month: 0.00 EUR	Cloud-to-device commands ⓘ Enabled
	IoT Edge ⓘ Enabled
	Device management ⓘ Enabled

Advanced Settings

Device-to-cloud partitions ⓘ 2

Review + create < Previous Basics Automation options





Microsoft Azure

Home > New > Storage account > Create storage account

Create storage account

Basics Networking Advanced Tags Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Azure for Students

Resource group * IoT

[Create new](#)

Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

Storage account name *

Location * (Europe) North Europe

Performance ☒ Standard ☐ Premium

Account kind ☐ Storage (general purpose v1)

Replication ☐ Locally-redundant storage (LRS)

[Review + create](#) < Previous Next: Networking >

Microsoft Azure

Home > New > Function App > Function App

Function App

Looking for the classic Function App create experience? ->

Basics Hosting Monitoring Tags Review + create

Create a function app, which lets you group functions as a logical unit for easier management, deployment and sharing of resources. Functions lets you execute your code in a serverless environment without having to first create a VM or publish a web application.

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Azure for Students

Resource Group * IoT

[Create new](#)

Instance Details

Function App name * Function App name.azurewebsites.net

Publish * Code Docker Container

Runtime stack * Node.js

Region * North Europe

[Review + create](#) < Previous Next: Hosting >

Microsoft Azure

Home > New > Function App > Function App

Function App

Looking for the classic Function App create experience? →

Basics | **Hosting** | Monitoring | Tags | Review + create

Storage

When creating a function app, you must create or link to a general-purpose Azure Storage account that supports Blobs, Queue, and Table storage.

Storage account *

(New) storageaccountiot905a
(New) storageaccountiot905a
dataa

Operating system

The Operating System has been recommended for you based on your selection of runtime stack.

Operating System *

Linux | **Windows**

Plan

The plan you choose dictates how your app scales, what features are enabled, and how it is priced. [Learn more](#)

Plan type *

Consumption

Review + create < Previous Next: Monitoring >

funktioo - Microsoft Azure

Home > All resources > funktioo

funktioo

Function Apps

"funktioo"

Azure for Students

Function Apps

funktioo

Functions

EventHubTrigger1

Integrate

Manage

Monitor

Proxies

Slots

+ New function

Functions

Search functions

NAME	STATUS
EventHubTrigger1	Enabled

